Theoretical Computer Science 72 (1990) 3-26
North-Holland

TOPOLOGICAL INVARIANTS OF CLASSIFICATION PROBLEMS

Lothar BUDACH
Humboldt-Universität, Sektion Mathematik, Postfach 1297, 1086 Berlin, GDR

Abstract. There is a general agreement that problems which are highly complex in any naive sense are also difficult from the computational point of view. It is therefore of great interest to find invariants and invariant structures which measure in some respect the complexity of the given problem. The question which we are going to consider in the following paper are classification problems, the "computations" are described by questionnaires [3, 10] or, as they are called nowadays, by "branching programs" [11]. The "complexity" of the problem is measured by classical topological invariants (Betti numbers, Euler-Poincaré characteristic) of topological structures (simplicial complexes, topological spaces).

1. Information systems

1.1. An *information system* (see [9]) is a quadruple $S=(X, A, V, \rho)$ where X, A, V are finite sets and ρ is a mapping of $X\times A$ into V. The set X is interpreted as the set of all *objects* under consideration, A is the set of all *attributes*, and V is the set of *descriptors*. The mapping ρ is the so-called *information function*. ρ defines two adjoint functions,

$$\bar{\rho}: X\to \mathrm{Hom}(A, V), \qquad \tilde{\rho}: A\to \mathrm{Hom}(X, V)$$

($\mathrm{Hom}(X, V)$ denotes the set of all mappings of X into V) defined in the usual way by $\bar{\rho}(x)(a):=\tilde{\rho}(a)(x):=\rho(x, a)$. We assume that $\tilde{\rho}$ is injective, i.e., different attributes define different functions $X\to V$. This enables us to identify the attribute a with the corresponding function and to write $a(x)$ instead of $\tilde{\rho}(a)(x)=\rho(x, a)$. Let $\mathrm{Im}\, a=\{a(x)\colon x\in X\}$ be the image of the function a, then a can be considered as a function of X onto $\mathrm{Im}\, a$. By abuse of language we consider A to be the set of functions $a: X\to \mathrm{Im}\, a$ and write $S=(X, A)$ instead of $S=(X, A, V, \rho)$.

Let $f: X\to Y$ be a mapping. We call f to be *dependent on* S if the following condition is satisfied:

$$\text{If } x_1, x_2\in X \text{ and } a(x_1)=a(x_2) \text{ for all } a\in A, \text{ then } f(x_1)=f(x_2).$$

The mapping f is dependent on S iff there is a function $\prod_{a\in A} \mathrm{Im}\, a\to Y$, such that the following diagram is commutative:

$$\begin{array}{ccccc} x & \in & X & \xrightarrow{f} & Y \\ \downarrow & & \downarrow & \nearrow & \\ (a(x))_{a\in A} & \in & \prod_{a\in A} \operatorname{Im} a & & \end{array}$$

A triple $C=(X, A, f)$ such that (X, A) is an information system, and $f: X \to Y$ is a function dependent on (X, A) is called a *classification problem*. The function f is called the *classifying function* or the *classification*. By some technical reasons we will assume throughout the following paper, that all information systems satisfy the following condition:

S is *fully faithful*, i.e., the function $X \to \prod_{a\in A} \operatorname{Im} a$ is bijective. This implies that all functions $f: X \to Y$ are dependent on S.

1.2. Examples

1.2.1. Every boolean function $f:\{0,1\}^n \to \{0,1\}$ defines a classification problem. The underlying information system $B_n = (X, A, V, \rho)$ is called the *boolean information system of order n*. It consists of $X=\{0,1\}^n$ as set of objects, $A=\{1,2,\ldots,n\}$ as set of attributes and $V=\{0,1\}$ as set of descriptors. $\rho: X\times A \to V$ is the selection function defined by $\rho((x_1,\ldots,x_n), i)=x_i$, and f is the classifying function.

1.2.2. Let Σ be a finite alphabet and L be a language over Σ. L is a subset of Σ^* and defines a subset $L^n := L\cap\Sigma^n$ for any natural number n. Let $f_n:\Sigma^n \to \{0,1\}$ be the characteristic function of L_n, i.e.,

$$f_n(w) := \begin{cases} 1, & \text{if } w \in L^n, \\ 0, & \text{otherwise.} \end{cases}$$

Therefore L defines for every n a classification problem with information function f_n and the underlying information system $(\Sigma^n, \{1,2,\ldots,n\}, \Sigma, \rho)$ with $\rho((\sigma_1,\ldots,\sigma_n), i)=\sigma_i$. For $\#(\Sigma)=2$ this information system is isomorphic to B_n.

1.2.3. Let V be a finite set, called the set of vertices and let v_0, v_e be two distinguished vertices of V. Take $V' = V-\{v_e\}$. A *V-maze* is a function $d: V'\times\{0,1\}\to V$ (see [1, 3, 12]). A V-maze can be considered as a directed graph $\Gamma(d)$, V being the set of vertices and $E=\{(v, d(v,i)): v\in V', i\in\{0,1\}\}$ the set of edges. This graph has two distinguished nodes v_0 and v_e. The outdegree of all nodes different from v_e is two and v_e is a sink of this graph, its outdegree being zero. The V-maze d is called to be threadable, if there is a path in $\Gamma(d)$ connecting v_0 with v_e. In other terms: a V-maze d is a finite automaton with input set $\{0,1\}$. v_0 is the initial state, v_e is the terminal state and d is threadable iff the language of this automaton is not empty.

Let $X=\operatorname{Hom}(V'\times\{0,1\}, V)$ be the set of V-mazes. These are the objects of the following information system:

- $A = V'\times\{0,1\}$ is the set of attributes,

- V is the set of descriptors, and
- $\rho : X \times A \to V$ is the function defined by $\rho(d, (v, i)) := d(v, i)$.

MAZES$(V) := (X, A, V, \rho)$ is an information system which is the underlying information system of the classification problem **GAP**(V, v_0, v_e) the classifying function t of which is defined by

$$t(d) := \begin{cases} 1, & \text{if } d \text{ is threadable,} \\ 0, & \text{otherwise.} \end{cases}$$

Since all **GAP**(V, v_0, v_e) with $\#(V) = n$ are isomorphic we write **GAP**(n) instead of **GAP**(V, v_0, v_e). Without loss of generality we can assume $V = \{1, 2, \ldots, n\}$ and $v_0 = 1$, $v_e = n$.

2. Questionnaires

2.1. Procedures to classify via a given classifying function are the questionnaires introduced by Picard [10] (see also [3-5]). A *questionnaire* or a *classifying graph* over a given information system $S = (X, A)$ is a quintuple $F = (Q, Y, \alpha, \delta, q_0)$ where

- Q is a finite set, the set of nodes;
- Y is a finite set with $Y \cap A = \emptyset$ and $\alpha : Q \to Y \cup A$ is a mapping; the nodes of act $(F) := \alpha^{-1}(A)$ are called questions and the nodes of term$(F) := \alpha^{-1}(Y)$ are the results;
- $\delta = \{\delta_q : q \in \text{act}(F)\}$, $\delta_q : \text{Im}\, \alpha(q) \to Q$ describes the strategy of posing questions;
- q_0 is the initial node, i.e., that node in which all enquiries get started.

In case $S = (X, A)$ is a boolean information system B_n, F is called a *branching program*. X operates partially on Q by the following function:

$$\begin{aligned} \text{act}(F) \times X &\to Q, \\ (q, x) &\mapsto qx := \delta_q(\alpha(q)(x)). \end{aligned}$$

This action can be interpreted as follows: Let q be a question, then $\alpha(q)$ is an attribute, i.e., a mapping $\alpha(q) : X \to V$. To pose question q to the object $x \in X$ means to apply $\alpha(q)$ on x. The answer of x to the question q is $\alpha(q)(x)$. This answer implies a new question or a result, namely $\delta_q(\alpha(q)(x))$ which we call qx. The partial action of X on Q can be extended to X^*, the free monoid generated by X. For $x \in X$ take $qx^{n+1} := (qx^n)x$ if $qx^n \in \text{act } F$. The sequence $q_0, q_0x, q_0x^2, \ldots, q_0x^m, \ldots,$ describes the strategy of F asking questions: after having asked for the attribute $\alpha(q_0x^m)$, $m < n$, F gets the answer $(\alpha(q_0x^m))(x)$ which makes F move to the node q_0x^{m+1}. There are two possibilities:

- There is an n with $q_0x^n \in \text{term}(F)$. Define in this case $n(x) := n$.
- $q_0x^n \in \text{act}(F)$ for all n. In this case we define $n(x) := \infty$.

Let $\xi_F : X \to Y$ be the following function,

$$\xi_F(x) := \begin{cases} \alpha(q_0 x^n), & \text{if } q_0 x^n \in \operatorname{term}(F), \\ \text{not defined}, & \text{otherwise.} \end{cases}$$

If F is free of cycles (more precisely if the directed graph

$$\Gamma(F) = (Q, \{(q, \delta_q(i)) : q \in \operatorname{act}(F), i \in \operatorname{Im} \alpha(q)\})$$

is free of cycles), then $qx^n \neq q$ for all $q \in \operatorname{act}(F)$, $x \in X$. In this case $n(x) \neq \infty$ for all $x \in X$ and, therefore, ξ_F is fully defined on X. So if we assume that F is free of cycles, then ξ_F is a fully defined function, which can be proved to be always (i.e., also in case when we do not assume that S is fully faithful) dependent on S. We say that F is a solution of a classification problem $C = (X, A, f)$ if F is free of cycles and if $\xi_F = f$.

2.2. It is easy to verify that every classification problem admits a solution F which is moreover a tree (for the easy proof of this fact we refer the reader to [3]). In [3, 4] we introduced different measures for the complexity of classifying graphs. One of these measures was the size of F:

$$\operatorname{size}(F) = \#(\operatorname{act}(F)).$$

Let $C = (X, A, f)$ be any classification problem. We introduce two numbers:

$$\operatorname{size}(C) = \min\{\operatorname{size}(F) : \xi_F = f \text{ and } F \text{ is free of cycles}\},$$

and

$$\operatorname{Size}(C) = \min\{\operatorname{size}(F) : \xi_F = f \text{ and } \Gamma(F) \text{ is a tree}\}.$$

One of the most interesting and outstanding problems in theoretical computer science is the determination of the "small size" $\operatorname{size}(C)$ of certain classification problems C. Though also the determination of $\operatorname{Size}(C)$ is not easy, it can be done in certain cases. The following section presents some results concerning these questions. The detailed proofs of these results can be read in [3].

3. Questionnaires for GAP(n)

3.1. Assume $Y = \{0, 1\}$ and let F_1, F_2 be classifying graphs with

$$F_i = (Q_i, Y, \alpha_i, \delta_i, q_{i0}), \quad i = 1, 2.$$

We define $F_1 \wedge F_2$ and $F_1 \vee F_2$ by the diagrams of Fig. 1.

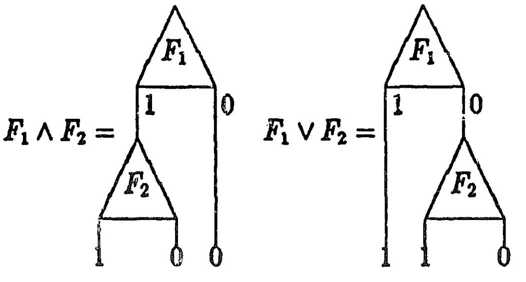

Fig. 1. Boolean operations of questionnaires.

Obviously,

$$\operatorname{size}(F_1 \wedge F_2) = \operatorname{size}(F_1 \vee F_2) = \operatorname{size}(F_1) + \operatorname{size}(F_2)$$

and

$$\xi_{F_1 \wedge F_2} = \xi_{F_1} \wedge \xi_{F_2}, \qquad \xi_{F_1 \vee F_2} = \xi_{F_1} \vee \xi_{F_2}.$$

3.2. In 1.2.3 we introduced already the information system **MAZES**(V) and the classification problem **GAP**(V, v_0, v_e). The latter can be considered as a special case of the following classification problem. Call a V-maze, d, k-threadable for a given natural number k, if it is threadable and if the path connecting v_0 with v has a length smaller or equal to k. Let $t_k = t_k(V, v_0, v_e)$ be the characteristic function of the set of all k-threadable mazes, i.e.,

$$t_k(d) := \begin{cases} 1, & \text{if } d \text{ is } k\text{-threadable}, \\ 0, & \text{otherwise.} \end{cases}$$

Let $\mathbf{GAP}_k(V, v_0, v_e)$ be the corresponding classification problem. Obviously $\mathbf{GAP}(V, v_0, v_e) = \mathbf{GAP}_k(V, v_0, v_e)$ with $k = \#(V) - 1$. It is easy to verify that one has the following equality:

$$t_{k+l} = (\bigvee \{t_k(V, v_0, v) \wedge t_l(V, v, v_e) \colon v \in V,\ v \neq v_0,\ v \neq v_e\}) \vee t_1.$$

From this equation one gets the following recursive inequality for $s(k, n) := \operatorname{size}(\mathbf{GAP}_k(V, v_0, v_e))$ with $n := \#(V)$:

$$s(1, n) = 2,$$

$$s(k+l, n) \leq \left(\sum_{i=2}^{n-1} s(k, n) + s(l, n)\right) + 2$$

$$= (n-2)(s(k, n) + s(l, n)) + 2.$$

From this formula results:

$$s(2k, n) \leq 2(n-2)s(k, n) + 2 \leq 2(n-1)s(k, n)$$

and this gives the following upper bound for $s(n) := \operatorname{size}(\mathbf{GAP}(n))$:

$$s(n) \leq 4n(n-1)^{\log n}.$$

3.3. Let us give an example: An optimal solution of the classification problem **GAP**(3), i.e., a classifying graph for **GAP**(3) with the minimal number $s(3) = 7$ of nodes, is the classifying graph described in Fig. 2.

3.4. The example 3.3 is the easiest case of the following solution F for **GAP**(n) which is in general not optimal but better than that of 3.2 for n small:

$$\operatorname{act}(F) = \{(U, n) \in 2^V \times N \colon 1 \in U,\ n \notin U,\ \#(U) - 1 \leq n < 2\#(U)\},$$

$$\operatorname{term}(F) = \{0, 1\} = Y,$$

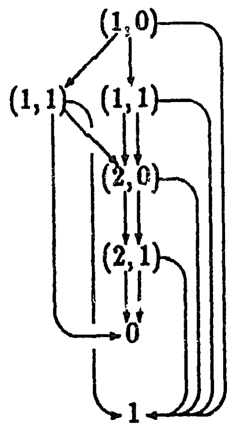

Fig. 2. An optimal solution for **GAP(3)**.

$\alpha(U, n) :=$ nth element of $U \times 2$ in lexicographical order,

$$\delta_{(U,n)}(y) = \begin{cases} (U \cup \{y\}, n+1), & \text{if } n+1 < 2\#(U \cup \{y\}) \text{ and } y \neq v_e, \\ 1, & \text{if } y = v_e, \\ 0, & \text{if } n+1 = 2\#(U \cup \{y\}), \end{cases}$$

$$q_0 = (\{v_0\}, 0).$$

It is easy to see, that

$$\text{size}(F) = 2 + \sum_{i=0}^{n-2} \binom{n-2}{i}(i+2) = (n-2)2^{n-3}.$$

For $n = 3$ we get $\text{size}(F) = 5$ and $\text{size}(F) = 12$ for $n = 4$. The first value is optimal and we believe that also $s(4) = 12$. But already the proof of this fact seems to be hard. The importance of the numbers $s(n)$ is demonstrated by the following theorem:

Theorem 3.5. *Let* L *be the class of all languages, which can be recognized by a Turing machine with logarithmic tape and let* NL *be the class of all languages which can be recognized by a nondeterministic Turing machine with logarithmic tape. Obviously* $\mathrm{L} \subseteq \mathrm{NL}$. *In order that* $\mathrm{L} = \mathrm{NL}$ *it is necessary that* $s(n)$ *is polynomial in n.*

The proof of this theorem can be found in [3].

4. Classifying trees

4.1. Let $S = (X, A)$ be an (fully faithful) information system. Let trees(S) be the set of all classifying trees over the given information system S. To every attribute $a \in A$ with $\mathrm{Im}\, a = \{y_1, \ldots, y_n\}$ there is an n-ary function $a : \text{trees}(S)^n \to \text{trees}(S)$

which is defined in the following way: Let $F_1, \ldots, F_n$ be elements of trees(S) with

$$F_i = (Q_i, Y_i, \alpha_i, \delta_i, q_{i0}),$$

then $F := a(F_1, \ldots, F_n) = (Q, Y, \alpha, \delta, q_0)$ is defined as follows:

$$Q = \{a\} \cup \left(\bigcup_{i=1}^{n} Q_i \times \{i\} \right), \qquad Y = \bigcup_{i=1}^{n} Y_i,$$

$$\alpha(a) := a, \qquad \alpha(q, i) := \alpha_i(q).$$

Then we get $\operatorname{act}(F) = \{a\} \cup (\bigcup_{i=1}^{n} \operatorname{act}(F_i) \times \{i\})$. The family of functions δ is defined by $\delta_{(q,i)}(y) := (\delta_q(y), i)$ and $\delta_a(i) := (q_{i0}, i)$. The initial node q_0 of F is defined by $q_0 := a$, which completes the definition of F.

Suppose we are given a fixed set Y. To every element $y \in Y$ we define the following trivial classifying tree:

$$[y] := (\{y\}, \{y\}, 1_{\{y\}}, \emptyset, \{y\}),$$

consisting of one node only. For the next theorem we consider by set theoretic reasons only classifying trees $F = (Q, Y, \alpha, \delta, q_0)$ with a fixed set Y of possible results of the classification.

Theorem 4.2. (trees(S), A) *is a free algebra and the set of all* $[y]$ *forms a set of free generators.*

The proof of this theorem is given in [3].

4.3. Consider the following binary relation $\sim$ on the set of all classifying trees.

(1) For $a \in A$ and $y \in Y$, $a([y], \ldots, [y]) \sim [y]$, holds.

(2) If $a, b \in A$, then

$$a(b(F_{11}, \ldots, F_{1n}), \ldots, b(F_{m1}, \ldots, F_{mn}))$$
$$\sim b(a(F_{11}, \ldots, F_{m1}), \ldots, a(F_{1n}, \ldots, F_{mn})).$$

(3) Suppose $t = a(t_1, \ldots, t_n)$ and t_i contains a subtree $t' = a(t'_1, \ldots, t'_n)$. Let $t - t'$ be the tree arising from t by replacing t' by t'_i, then $t \sim t - t'$.

Definition 4.4. The congruence relation $\equiv$ generated by $\sim$ will be called the syntactic congruence of classifying trees.

Consider the example described in Fig. 3.

4.5. Two classifying trees F_1 and F_2 will be called semantically equivalent, $F_1 \approx F_2$, if their classifying functions are identical: $\xi_{F_1} = \xi_{F_2}$. It is obvious that classifying trees which are syntactically equivalent are semantically equivalent too. More interesting is the other direction which will be the main result of the following theorem, the proof of which will also be found in [3].

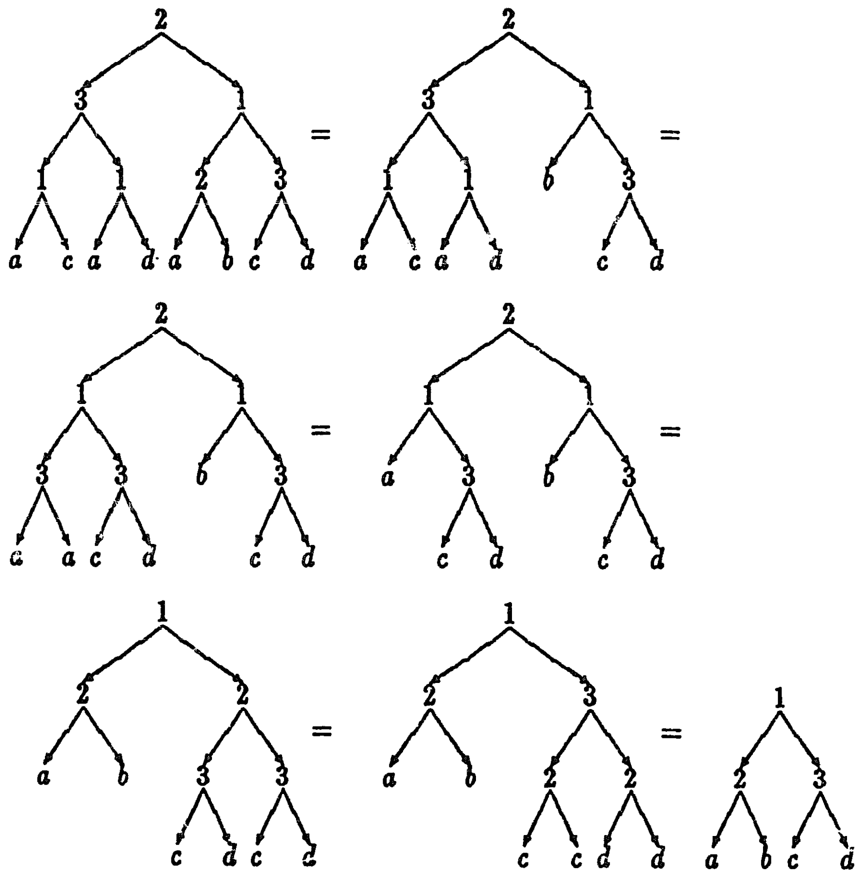

Fig. 3. Transformation of decision trees.

Theorem 4.6. *Syntactical and semantical equivalence are equal, i.e., for all classifying trees the following holds:* $F_1 \approx F_2$ *if and only if* $F_1 \equiv F_2$.

4.7. *Warning.* For this theorem the assumption that S is fully faithful is of significance. See [3, Example 4.13].

5. Optimal trees for GAP(n)

5.1. In 2.2 we introduced the notion of the "big size" of a classification problem $C=(X, A, f)$:

$$\mathrm{Size}(C) := \min\{\mathrm{size}(F)\colon \xi_F = f \text{ and } \Gamma(F) \text{ is a tree}\}.$$

A classifying tree F is called an optimal tree solution of C or for short an optimal tree for C if F is a solution of C and if moreover $\mathrm{size}(F) = \mathrm{Size}(C)$. Since every classification problem has a solution which is a tree, every classification has an optimal tree solution. The example at the end of 4.4 gives evidence that it might not be easy to find an optimal tree for C and moreover it may be rather difficult to decide whether a given classifying tree is an optimal solution. We will find in 7.7 a criterion which allows the answer to the second question in certain cases. Let us first consider the problem **GAP**(n). Define $\sigma(n) := \text{Size}(\mathbf{GAP}(n))$. We intend now to give a recursion which allows to compute these numbers and to give lower bounds for $\sigma(n)$.

5.2. Let V be a finite set with two distinguished elements v_0 and v_e. In 1.2.3 we introduced V-mazes as functions $d: V' \times \{0, 1\} \to V$ with $V' := V - \{v_e\}$. A partial V-maze is a partial function $d: V' \times \{0, 1\} \supseteq\!\!\to V$. As for V-mazes we can consider the directed graph $\Gamma(d)$ with V being the set of vertices of $\Gamma(d)$ and E, the set of edges, being defined by

$$E := \{(v, d(v, i)) : (v, i) \in \text{dom } d\}.$$

Let d be a partial V-maze. Let reach d be the set of all $v \in V$ which are reachable from v_0 by a path from v_0 to v. d is called *trunk* if dom $d \subseteq (\text{reach } d) \times \{0, 1\}$ and a *complete trunk* if equality holds in this inclusion. d is *treadable* if $v_e \in$ reach d. d is called to be *disconnected* if it is not threadable and *stably disconnected* if all extensions $d' \supseteq d$ are disconnected. Let d be a partial maze. Define

$$d'' = d \downarrow ((\text{reach } d) \times \{0, 1\}) \cap \text{dom } d.$$

Obviously d'' is a trunk and it is easy to see that d is stably disconnected if and only if d'' is a complete disconnected trunk. Let F be a tree solution of **GAP**(V, v_0, v_e) and let q be any node of F. Let

$$q_0 \xrightarrow{x_0} q_1 \xrightarrow{x_1} q_2 \xrightarrow{x_2} \cdots \xrightarrow{x_{n-1}} q_n$$

be the path in F connecting the initial node q_0 of F with q. Since

$$\alpha(q_t) \in A = V' \times \{0, 1\},$$

we have $\alpha(q_t) = (v_t, i_t)$. Consider the set $\{((v_t, i_t), x_t) : t = 0, 1, \ldots, n-1\}$ which we call d_q.

Theorem 5.3. *If F is an optimal tree for* **GAP**(V, v_0, v_e), *then d_q is the graph of a partial function $d_q: V' \times \{0, 1\} \supseteq\!\!\to V$, i.e., d_q is a partial maze. Moreover*:

(1) *d_q is a trunk for all nodes q of F.*
(2) *For all $q \in \text{act}(F)$, d_q is neither complete nor threadable.*
(3) *For all $q \in \text{term}(F)$ with $\alpha(q) = 1$, d_q is threadable.*
(4) *For all $q \in \text{term}(F)$ with $\alpha(q) = 0$, d_q is complete and disconnected, i.e., stably disconnected.*

5.4. Every partial maze d defines a point in the grid $\mathbb{N}^2$ by

$$\eta(d) := (\#(\text{reach } d) + 1, \#(\text{dom } d) - \#(\text{reach } d) + 1).$$

Every node q of an optimal tree for **GAP**(V, v_0, v_e) defines a point $\eta(q) := \eta(d_q)$

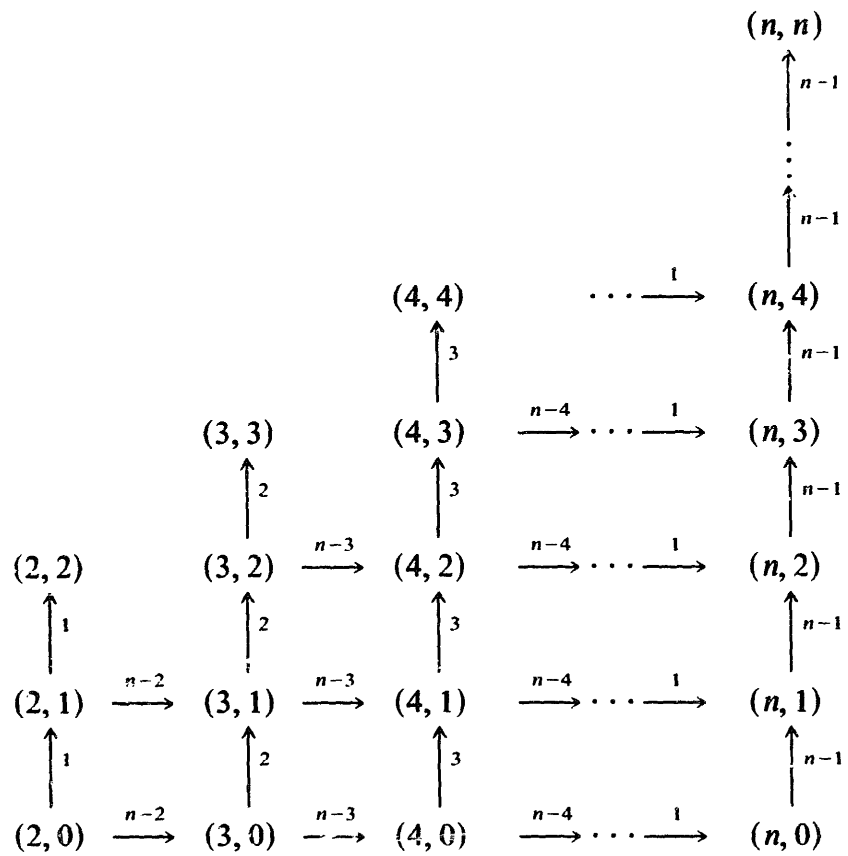

Fig. 4. The graph A^n.

of $\mathbb{N}^2$. Consider the following graph $A^n = (V^n, E^n)$:

$$V^n := \{(x, y) \in \mathbb{N}^2 \colon 0 \leq y \leq x,\ 2 \leq x \leq n\} \cup \{1, 2, \ldots, 2(n-1)\},$$

$$\begin{aligned} E^n := \{((x, y), i, (x+1, y)) \colon i \in \{1, 2, \ldots, n-x\}\} \cup \\ \{((x, y), i, (x, y+1)) \colon i \in \{1, 2, \ldots, x-1\},\ y < x\} \cup \\ \{((x, y), 1, x+y-1) \colon y < x\}. \end{aligned}$$

Consider the pictorial representation of A^n in Fig. 4 where we have omitted the edges of the third kind between (x, y) and $x+y-1$:

Theorem 5.5. *If F is an optimal solution for* GAP(n), *then F is a covering tree of the graph A^n. This implies that $\sigma(n) = \mathrm{size}(F)$ is equal to the number of simple paths in A^n.*

Corollary 5.6. $\sigma(n) = \Omega(n^n (n-2)!)$.

For details we refer the reader to [3].

6. Arsenals

6.1. A finite poset P is said to be *homogenous* (sometimes also called *pure*) if all maximal chains have the same length. A homogenous poset satisfies the *Jordan–Dedekind condition*: if x and y are two elements and if $x<y$, then $P_{\leqslant x}:=\{z: z\leqslant x\}$, $P_{\geqslant x}:=\{z: z\geqslant x\}$ and $[x,y]:=\{z: x\leqslant z\leqslant y\}$ are homogenous.

The symbol "$\lhd$" denotes the *covering relation*: $x\lhd y$, if $x<y$ and if $x<z\leqslant y$ implies $z=y$. If P is a finite homogenous poset then a *rank function* $r: P\to\mathbb{N}$ can be defined as follows:

(1) If P has a least element $\mathbf{0}$, then we define $r(\mathbf{0}):=0$, otherwise we define $r(x):=1$ for all minimal elements x.

(2) If $x\lhd y$, then $r(y):=r(x)+1$.

Definition 6.2. An *arsenal* is a finite homogenous poset having a maximal element $\mathbf{1}$ and a minimal element $\mathbf{0}$ which is the underlying set Γ of vertices of a directed graph (Γ, E, o, t). As usual we assume, that o and t are functions from the set E of edges into the set Γ of vertices. $o(e)$ is the beginning and $t(e)$ is the end of e. Moreover we assume that E is divided into two disjoint subsets $E=E_{\mathrm{a}}\cup E_{\mathrm{b}}$, the elements of E_{a} are called *amalgamations* and the elements of E_{b} *branchings*. By abuse of language we call the arsenal, defined by $(\Gamma, E_{\mathrm{a}}, E_{\mathrm{b}}, o, t)$ also Γ. The following conditions are to be satisfied:

(A1) If $e\in E_{\mathrm{a}}$, then $o(e)<t(e)$.

(A2) If $x,y\in\Gamma$, $x\lhd y$, then there is an $e\in E_{\mathrm{a}}$ with $o(e)=x$, $t(e)=y$.

(A3) If $e\in E_{\mathrm{b}}$, then $t(e)\leqslant o(e)$.

(A4) For any $x\in\Gamma$, $x\neq\mathbf{0}$ there is an $e\in E_{\mathrm{b}}$ with $o(e)=x$, $t(e)<x$.

Definition 6.3. A *homomorphism* of the arsenal $\Gamma^1=(\Gamma, E_{\mathrm{a}}^1, E_{\mathrm{b}}^1, o^1, t^1)$ into the arsenal $\Gamma^2=(\Gamma^2, E_{\mathrm{a}}^2, E_{\mathrm{b}}^2, o^2, t^2)$ is by definition a triple $(\Phi, \Phi_{\mathrm{a}}, \Phi_{\mathrm{b}})$ consisting of functions $\Phi:\Gamma^1\to\Gamma^2$, $\Phi_{\mathrm{a}}: E_{\mathrm{a}}^1\to E_{\mathrm{a}}^2$, $\Phi_{\mathrm{b}}: E_{\mathrm{b}}^1\to E_{\mathrm{b}}^2$ and satisfying the following conditions:

(H1) Φ is a homomorphism of the poset Γ_1 into the poset Γ_2.

(H2) $o_2\Phi_s=\Phi o_1$ and $t_2\Phi_s=\Phi t_2$ for $s\in\{\mathrm{a},\mathrm{b}\}$.

6.4. Remarks

6.4.1. Let $e\in E$ be a branching. Then $r(t(e))\leqslant r(o(e))$. The difference $r(e):=r(o(e))-r(t(e))$ is called the *degree* of the branching. Branchings of degree zero will be called *trivial*. An arsenal is called a *boolean arsenal* if the degree of all branchings is bounded by one.

Let Γ be a boolean arsenal and let E_{b}' denote the subset of all nontrivial branchings. There is an application $-: E_{\mathrm{b}}'\to E_{\mathrm{a}}$ with $o(\bar{e})=t(e)$, $t(\bar{e})=o(e)$. $\bar{e}$ is called an *inverse amalgamation* of the branching e.

Proof. Suppose $e \in E_b'$. Then $t(e) \leq o(e)$. Since e is nontrivial, $r(e) = 1$ and therefore $t(e) \lhd o(e)$. By (A2) there is an amalgamation $\bar{e}$ with $o(\bar{e}) = t(e)$, $t(\bar{e}) = o(e)$.

6.4.2. In general, one has the following result:

For any branching e there is a path from $t(e)$ to $o(e)$ the edges of which are amalgamations only. This means that to any branching there is an inverse sequence of amalgamations.

This is true because $t(e) \leq o(e)$. Therefore $t(e) = o(e)$ or

$$t(e) = x_1 \lhd x_2 \lhd \cdots \lhd x_n = o(e).$$

Due to (A2) we get the result.

6.4.3. (Γ, E, o, t) is strongly connected, i.e. for every pair of points x, y there is a path

$$x = x_1 \xrightarrow{e_1} x_2 \xrightarrow{e_2} \cdots \xrightarrow{e_{n-1}} x_n = y.$$

Proof. Using (A4) we get a path

$$x = x_1 \xrightarrow{e_1} x_2 \xrightarrow{e_2} \cdots \xrightarrow{e_{m-1}} x_m = \mathbf{0}$$

with $e_i \in E_b$ for $i = 1, 2, \ldots, m-1$. Since $\mathbf{0} \leq y$, (A2) gives the result.

6.5. Consider the free category C defined by the graph (Γ, E, o, t). The objects of this category are the elements of Γ, the morphisms $p: x \to y$ are the paths connecting x with y. Consider the following two functions w, $|\,|: E_a \cup E_b \to \mathbb{N}$:

$$w(e) := \begin{cases} 1, & \text{if } e \in E_b, \\ 0, & \text{otherwise,} \end{cases}$$

and $|e| := 1$. w and $|\,|$ can be extended to functors w, $|\,|: C \to \mathbb{N}$ in a unique way. It is obvious that, for any path $p: x \to y$, $w(p) \leq |p|$ holds.

6.6. Let x, y be two elements of Γ. The information of y with respect to x is defined as follows:

$$I(y \mid x) := \min\{w(p) \colon p: x \to y\}$$

and the *distance of y from x* by

$$d(x, y) := \min\{|p| \colon p: x \to y\},$$

where $p: x \to y$ is a path from x to y. $I(x) := I(x \mid \mathbf{1})$ is called the information and $d(x) := d(\mathbf{1}, x)$ is called the distance of x. The following properties are obviously true:

(I1) $I(y \mid x) = 0$ if $x \leq y$.

(I2) $I(y \mid x) \leq I(y) \leq I(\mathbf{0})$ for all $x, y \in \Gamma$.

(I3) $I(y \mid x) \leq I(z \mid x) + I(y \mid z)$ and $d(x, y) \leq d(x, z) + d(z, y)$.

(I4) $I(y \mid x) \leq d(x, y)$. Hence $I(x) \leq d(x) \leq I(x)^2$.

6.7. As usual we are representing graphs in the following manner: vertices are represented by points in the plane and edges are represented by arrows, connecting the corresponding points. For arsenals we assume the following additional laws to be respected: if $x<y$, then the point corresponding to x is on a lower level than the point corresponding to y. Therefore, arrows representing amalgamations have always a down-up direction. Trivial branchings will be omitted in the pictorial representation. In case of boolean arsenals every nontrivial branching has an inverse amalgamation. Hence branchings and their inverses can be considered like edges in a non-directed graph and this leads us to represent them by lines without an orientation given by an arrow. It is always clear, which of the two directions is the branching and which the amalgamation: the branching goes always down and the inverse amalgamation up. Consider the two examples of boolean arsenals described in Fig. 5. For b_1 we get, obviously, $I(\mathbf{1})=0$, $I(\mathbf{0})=1$ and for b_2 one obtains:

$$I(\mathbf{1})=0,$$
$$I(z)=1,$$
$$I(v)=I(y)=2,$$
$$I(\mathbf{0})=I(u)=I(x)=3.$$

6.8. Let Γ be an arsenal and let G be the automorphism group of Γ. Let gx be the image of x, if we apply $g\in G$ and let $Gx=\{gx: g\in G\}$ be the orbit of x. $\bar{\Gamma}=(\bar{\Gamma}, \bar{E}_a, \bar{E}_b, \bar{o}, \bar{t})$ is by definition the following arsenal: its vertices are the orbits. If Gx, Gy are two orbits, then we define: $Gx<Gy$ if there is a $g\in G$ with $x<gy$. In this way the set Γ/G of orbits becomes a poset and it is easy to verify, that $Gx\lhd Gy$ iff there is a $g\in G$ with $x\lhd gy$. Therefore we define:

$$\bar{E}_a:=\{(Go(e), Gt(e)): e\in E_a\}, \qquad \bar{E}_b:=\{(Go(e), Gt(e)): e\in E_b\}$$

and

$$o(Gx, Gy):=Gx, \qquad t(Gx, Gy):=Gy.$$

The canonical mapping $\Phi:\Gamma\to\bar{\Gamma}$ with $\Phi(x)=Gx$, $\Phi_s(e)=(Go(e), Gt(e))$, for $e\in E=E_a\cup E_b$, $s\in\{a,b\}$ is a homomorphism of Γ onto $\bar{\Gamma}$. It is easy to verify that $r(x)=r(\Phi(x))$, $d(x,y)=d(\Phi(x),\Phi(y))$, $I(x\,|\,y)=I(\Phi(x),\Phi(y))$.

6.9. Let $S=(X, A, V, \rho)$ be an information system which is assumed as always to be fully faithful. S defines an arsenal $\Gamma(S)=(\Gamma(S), E_a, E_b, o, t)$ in the following manner: $\Gamma(S)$ is the lattice of all equivalence relations on X. If C and D are equivalence relations, then $C\leq D$ if xCy implies xDy. In other words: every equivalence class of D is union of equivalence classes of C. $\mathbf{1}$ is the trivial equivalence

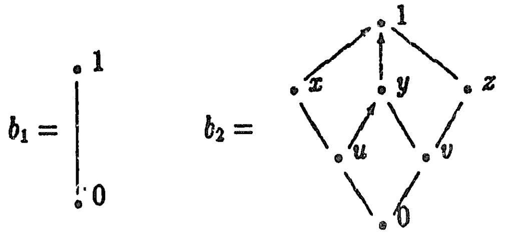

Fig. 5. The arsenals b_1 and b_2.

relation where all elements of X are equivalent and $\mathbf{0}$ is the identity relation. Let $a \in A$ be an attribute. It defines for every equivalence relation C and for every equivalence class E of C a branching $e_{(C,E,a)}$ with $o(e_{(C,E,a)}) = C$ and $D = t(e_{(C,E,a)})$ to be defined in the following manner:

(1) If $x \notin E$, then (xDy iff xCy).

(2) If $x \in E$, then (xDy iff xCy (i.e., $y \in E$) and $a(x) = a(y)$).

D will be denoted by $C +_E a$. So the class E is branched into subsets of equivalence classes of the equivalence relation Ker a. Because X is assumed to be fully faithful, condition (A4) of Definition 6.2 is satisfied because if C is not the identity relation then there is at least one class E with more than two elements which can be separated by at least one attribute. The arsenal $\Gamma(S)$ is called the *arsenal belonging to S* and $\overline{\Gamma(S)}$ is called the *reduced arsenal belonging to S*. If $S = B_n$ is the boolean information system defined in 1.2.1, then we call $\Gamma(B_n)$ the *boolean arsenal of order n*. Moreover we define $b_n := \overline{\Gamma(B_n)}$ and call b_n to be the *reduced boolean arsenal of order n*. The reduced boolean arsenals of orders 1 and 2 are represented in Fig. 4.

Remark that every classification problem $C = (X, A, f)$ over a given information system (X, A) defines an equivalence relation $\operatorname{Ker} f$ on X, which is an element of $\Gamma(S)$, and therefore an element $\ker f = \Phi(\operatorname{Ker} f)$ of the corresponding reduced arsenal. We will prove in the next proposition that $I(\operatorname{Ker} f) = I(\ker f)$ is a lower bound of the size of f in the sense of 2.2.

Proposition 6.10. $I(\operatorname{Ker} f) = I(\ker f) = \operatorname{size}(f)$.

For the proof of this proposition we need the following lemma:

Lemma 6.11. *In any questionnaire F which is free of cycles there is a question q with* $\delta_q(\operatorname{Im} \alpha(q)) \subseteq \operatorname{term}(F)$.

Proof. Suppose that for every $q \in \operatorname{act}(F)$ there is a $\nu_q \in \operatorname{Im} \alpha(q)$ with $\delta_q(\nu_q) \in \operatorname{act}(F)$. Define $\pi(q) := \delta_q(\nu_q)$. Then

$$q \xrightarrow{\nu_q} \pi(q) \xrightarrow{\nu_{\pi(q)}} \pi^2(q) \xrightarrow{\nu_{\pi^2(q)}} \cdots$$

is an infinite path in $\operatorname{act}(F)$ which contains a cycle because F is finite.

6.12. Let $F = (Q, Y, \alpha, \delta, q_0)$ be a questionnaire without cycles, $f = \xi_F$ and let q be a question with $\delta_q(\operatorname{Im} \alpha(q)) \subseteq \operatorname{term}(F)$. Let F' be the questionnaire which arises from F by making q to a terminal node, i.e. we add a new element y_0 to the set Y and define $\alpha'(q) := y_0$. Let all other parts of F be unchanged. Then $\xi_{F'}(x) := \alpha'(q_0 x'')$ if $q_0 x'' \in \operatorname{term}(F') = \operatorname{term}(F) \cup \{q\}$. Assume $f' = \xi_{F'}$. Then

$$f'(x) = \begin{cases} \xi_F(x) = f(x), & \text{if } q_0 x'' \neq q, \\ y_0, & \text{if } q_0 x'' = q. \end{cases}$$

Let E be the set of all x with $f'(x) = y_0$ and assume $a = \alpha(q)$. E is an equivalence class of $\operatorname{Ker} f'$. Hence $e_{(\operatorname{Ker} f', E, a)}$ is a branching in $\Gamma(S)$ with $o(e_{(\operatorname{Ker} f', E, a)}) = \operatorname{Ker} f'$

and $\mathrm{Ker}\, f' +_E a = t(e_{(\mathrm{Ker}\, f', E, a)}) = \mathrm{Ker}\, f$. We have proved so far that to any f there is an f' with

$$I(\ker f) \leq I(\ker f') + 1,$$

$$\mathrm{size}(f') \leq \mathrm{size}(f) - 1.$$

By induction one gets $I(\ker f) \leq \mathrm{size}(f)$. The other inclusion can be proved by constructing a questionnaire using the $I(\ker f)$ branchings as questions.

7. Conditions and colored posets

7.1. Simplicial complexes

A *finite simplicial complex* K is by definition a nonempty family of nonempty subsets called *simplexes* of a set $\{v\}$ of *vertices* such that

(1) any set consisting of exactly one vertex is a simplex,

(2) any nonempty subset of a simplex is a simplex.

For details we refer the reader to [13]. A simplex s which is contained in a simplex t is called a *face of* t. The *dimension* of a simplex s, $\dim s$ is the maximum of the dimensions of all simplexes of K. The maximal simplexes, i.e. those simplexes which are maximal under inclusion, are called *facets*. K is said to be *homogeneously n-dimensional* if every simplex is a face of an n-dimensional simplex. So in this case all facets are n-dimensional.

Every finite simplicial complex K defines a finite poset $(K, \subseteq)$ the elements of which are the simplexes of K and these are partially ordered by inclusion. If K is homogeneously n-dimensional, then the corresponding poset is homogeneous of rank $n+1$. Its rank function r satisfies obviously the following condition: $r(s) = \dim s + 1$.

Let P be an arbitrary poset. It defines a simplicial complex $\Delta(P)$ in the following way: The vertices of $\Delta(P)$ are the elements of P and the simplexes of $\Delta(P)$ are nonempty subsets $\{x_0, x_1, \ldots, x_p\}$ of P such that $x_0 < x_1 < \cdots < x_p$. If K is a simplicial complex, then $K' := \Delta(K)$ (K to be considered as a poset) is called the *barycentric subdivision of* K.

7.2. Conditions

7.2.1. Let $S = (X, A, V, \rho)$ be an information system with $N := \#A - 1$. We assume as usual that S is fully faithful. An *A-condition* is defined to be a partial function $c: A \supseteq\!\!\rightarrow V$ satisfying $c(a) \in \mathrm{Im}\, a$ for all $a \in \mathrm{dom}\, c$. As usual c can be considered as a subset of the product $A \times V$ via

$$c = \{(a, v)\colon a \in \mathrm{dom}\, c,\ v = c(a)\}.$$

Consider the following simplicial complex:

- The set of vertices is $A \times V$,
- the set $\mathbf{Cond}(S)$ of simplexes is the set of all S-conditions considered as subsets of $A \times V$.

The facets of $\mathbf{Cond}(S)$ are the fully defined functions $c: A \to V$. They all are of dimension N. Hence $\mathbf{Cond}(S)$ is a homogeneously N-dimensional simplicial complex.

7.2.2. If x is an arbitrary object of X, then $\bar{\rho}(x)$ is a facet, called the facet belonging to x. Let c be a condition and let x be an object of X. We say that x satisfies c if, for all $a \in \operatorname{dom} c$, $a(x) = c(a)$ holds. Let $\mathbf{Sat}(c)$ be the set of all objects which satisfy c. Obviously the following condition is true:

$$x \in \mathbf{Sat}(c) \text{ if and only if } c \subseteq \bar{\rho}(x).$$

7.2.3. Let $B_n = (X, A, V, \rho)$ be the boolean information system described in 1.2.1, $n = N$. $X = \{0, 1\}^n$ can be mapped one to one onto the set of vertices of the n-dimensional cube C_n. On the other side $|\mathbf{Cond}(B_n)|$, the geometric realization of $\mathbf{Cond}(B_n)$, is a convex polyhedron which is dual to C_n. In case $n = 3$, $|\mathbf{Cond}(B_3)|$ is the octahedron (see also Fig. 6). Every condition c represents a simplex of $\mathbf{Cond}(B_n)$, i.e., a face of $|\mathbf{Cond}(B_n)|$. We leave it as an exercise for the reader that $\mathbf{Sat}(c)$ is a subset of $\{0, 1\}^n$ which forms the set of vertices of the face of C_n which is dual to c. From this follows that c is uniquely defined by $\mathbf{Sat}(c)$ and that the simplicial complex $\mathbf{Cond}(B_n)$ considered as a poset is dual to the poset $\mathscr{F}(C_n)$ of faces of the n-dimensional cube C_n.

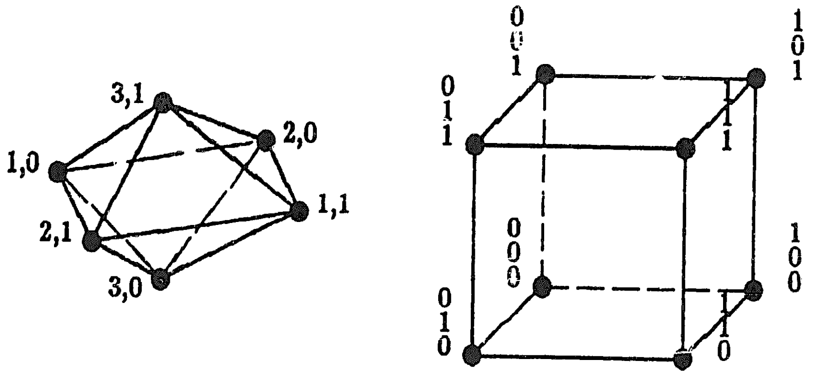

Fig. 6. $\mathbf{Cond}(B_3)$ and the three-dimensional unit cube.

7.2.4. There are different possibilities to describe conditions. Consider the boolean information system described in 1.2.1. Assume $N = n = 3$ and let c_1, c_2 be the conditions $c_1 = \{(2, 0), (3, 0)\}$, $c_2 = \{(1, 1), (2, 0)\}$. These conditions can be denoted as follows:

$$c_1 = \text{“}2 = 0\text{”} \wedge \text{“}3 = 0\text{”}, \qquad c_2 = \text{“}1 = 1\text{”} \wedge \text{“}2 = 0\text{”}.$$

In general we can write

$$c = \bigwedge_{a \in \operatorname{dom} c} \text{“}a = c(a)\text{”}.$$

If we assume the set A of attributes given in a certain order, $A = \{a_1, \dots, a_N\}$, and if we define

$$w_i := \begin{cases} c(a_i), & \text{if } a_i \in \operatorname{dom} c, \\ \bullet, & \text{otherwise,} \end{cases}$$

then $w := w_1 w_2 \dots w_N$ is a representation of the condition c. For c_1, c_2 we get: $c_1 = \bullet 00$, $c_2 = 10\bullet$.

7.2.5. Assume $S = B_n$ to be a boolean information system (see 1.2.1). Define for a given S-condition C the natural numbers ε_i ($i \in \operatorname{dom} c$) in the following way:

$$\varepsilon_i := \begin{cases} 1, & \text{if } c(i) = 1, \\ -1, & \text{if } c(i) = 0. \end{cases}$$

Let $x_i^1 = x_i$, $x_i^{-1} = \neg x_i = \overline{x_i}$. Then

$$\bigwedge_{i \in \operatorname{dom} c} x_i^{\varepsilon_i}$$

is called the *prime implicant description of* c. Consider an object $x = (x_1, \dots, x_n)$. The following equivalences are true:

$$\begin{aligned} \bigwedge_{i \in \operatorname{dom} c} x_i^{\varepsilon_i} = 1 \;&\Leftrightarrow\; x_i^{\varepsilon_i} = 1 \text{ for all } i \in \operatorname{dom} c \\ &\Leftrightarrow\; \text{for all } i \in \operatorname{dom} c, \\ &\qquad (x_i = 1 \text{ iff } c(i) = 1) \text{ and } (\neg x_i = 1 \text{ iff } c(i) = 0) \text{ holds} \\ &\Leftrightarrow\; \text{for all } i \in \operatorname{dom} c,\ x_i = c(i) \text{ holds} \\ &\Leftrightarrow\; c \subseteq \bar{\rho}(x) \\ &\Leftrightarrow\; x \in \mathbf{Sat}(c). \end{aligned}$$

Since c is completely defined by $\mathbf{Sat}(c)$, the prime implicant description gives a complete characterization of c and we write, by abuse of language,

$$c = \bigwedge_{i \in \operatorname{dom} c} x_i^{\varepsilon_i}.$$

For c_1 and c_2 we get:

$$c_1 = \bar{x}_2 \bar{x}_3, \qquad c_2 = x_1 \bar{x}_2.$$

7.2.6. Dually to the prime implicant description can be defined the *prime clause description of* c. Define ε_i as above, then

$$\bigvee_{i \in \operatorname{dom} c} x_i^{-\varepsilon_i}$$

is called the prime clause description of c. We get

$$\bigvee_{i \in \operatorname{dom} c} x_i^{-\varepsilon_i} = 0 \Leftrightarrow x \in \mathbf{Sat}(c).$$

Hence the prime clause description is also a complete description of c and we write, by abuse of language,

$$c = \bigvee_{i \in \operatorname{dom} c} x_i^{-\varepsilon_i}.$$

For c_1 and c_2 we get

$$c_1 = x_2 \vee x_3, \qquad c_2 = \bar{x}_1 \vee x_2.$$

7.3. Colored posets

Let P be a finite homogeneous poset and let $g : P \supseteq\!\!\rightarrow Y$ be a partial function. We will say that an element $a \in P$ is *colored*, if it is in the domain of g. The value $g(x)$ will be called the *color of* x. $g : P \supseteq\!\!\rightarrow Y$ is called a precoloring of P and (P, g, Y) is called a precolored poset if, the following conditions hold:

(1) All maximal elements of P are colored.

(2) If x is colored and $x < y$, then y is colored and has the same color.

g is called a *coloring* and (P, g, Y) is called a *colored poset* if in addition to properties (1) and (2) the following holds:

(3) If x is an element of P and if all y with $x < y$ are colored and have the same color, then x is also colored (and has, as a consequence of (2), the same color).

Let (P, g, Y) be a precolored poset. Let $\max(P)$ be the set of all maximal elements of P. Condition (3) is equivalent to either one of the following conditions:

(3′) If $x \in P$ and if all y with $x \lhd y$ are colored and have the same color, then x is also colored.

(3″) If $x \in P$ and if all y with $x < y$ and $y \in \max(P)$ have the same color, then x is colored.

Obviously every precoloring g can be extended in a unique manner to a coloring $\bar{g}$ by the following procedure:

> $x \in \operatorname{dom} \bar{g}$ iff all elements $y \in \max(P) \cap P_{\geq x}$ have the same color.
> If this color is y_0, then define $\bar{g}(x) := y_0$.

7.4. A subset U of P will be called an ascending subset (sometimes these subsets are also called open subsets, because they define a topology in P), if $x \in U$, $x < y$ implies $y \in U$. U is called *pure* if either one of the following conditions (which are equivalent to each other) is satisfied:

(1) If x is an element of P and if all y with $x < y$ are in U, then $x \in U$.

(2) If x is an element of P and if all y with $x \lhd y$ are in U, then $x \in U$.

(3) If x is an element of P and if all y with $x < y$ and $y \in \max(P)$ are in U, then $x \in U$.

Every subset U of P can be embedded into a smallest pure subset $\mathbf{Pure}(U)$ by the following procedure:

(1) Add all elements y of $\max(P)$ to $\mathbf{Pure}(U)$ for which there is an $x \in U$ with $x < y$.

(2) Add all elements y to $\mathbf{Pure}(U)$ with $\max(P) \cap P_{\geq y} \subseteq \mathbf{Pure}(U)$.

If $g: P \supseteq\!\!\rightarrow Y$ is a precoloring, then the following property is obviously satisfied:

$$\mathbf{Pure}(g^{-1}(y)) = g^{-1}(y) = \mathbf{Pure}(\bar{g}^{-1}(y)).$$

Let us define:

$$\mathbf{Pure}(g, y) := \mathbf{Pure}(g^{-1}(y)) = \bar{g}^{-1}(y),$$

$$\mathbf{Pure}(g) := \bigcup_{y \in Y} \mathbf{Pure}(g, y) = \operatorname{dom} \bar{g},$$

$$\mathbf{Mix}(g) := P - \mathbf{Pure}(g).$$

Then the following equalities hold:

$$P = \mathbf{Pure}(g) \cup \mathbf{Mix}(g),$$

$$\emptyset = \mathbf{Pure}(g) \cap \mathbf{Mix}(g).$$

Define the closure of $\mathbf{Pure}(g, y)$ to be

$$\overline{\mathbf{Pure}}(g, y) := \{c \in \mathbf{Cond}(S): c \subseteq d \in \mathbf{Pure}(g, y)\}.$$

7.5. $\mathbf{Pure}(g, y)$ and $\mathbf{Pure}(g)$ are pure subsets of P and therefore ascending. $\mathbf{Mix}(g)$ is a descending subset of P. Therefore in case that P is a simplicial complex, $\mathbf{Mix}(g)$ is a subcomplex of P but $\mathbf{Pure}(g, y)$ and $\mathbf{Pure}(g)$ are not. This leads us to the following definition: If P is a simplicial complex and if g is a coloring of P (where P is considered as a poset) then we define:

$$\mathbf{Pure}(g, y) := \Delta(\mathbf{Pure}(g, y)),$$

$$\mathbf{Pure}(g) := \Delta(\mathbf{Pure}(g)).$$

This definition forces $\mathbf{Pure}(g, y)$ and $\mathbf{Pure}(g)$ to be simplicial complexes.

Example 7.6. Let $K = (S, Y, f)$ be a classification problem and let $S = (X, A, V, \rho)$ be the underlying information system, which is assumed as always to be fully faithful. As we have already seen in 7.2, S defines a simplicial complex $\mathbf{Cond}(S)$. The maximal elements of $\mathbf{Cond}(S)$ are the facets of $\mathbf{Cond}(S)$ which in turn are the elements of $\prod_{a \in A} \operatorname{Im} a$. Therefore we can define the function

$$g: \max(\mathbf{Cond}(S)) \rightarrow \prod_{a \in A} \operatorname{Im} a \rightarrow X \xrightarrow{f} Y,$$

i.e., a precoloring of $\mathbf{Cond}(S)$ which defines in turn a coloring $\bar{g}$ which we denote by f^*. $\mathbf{Mix}(f) := \mathbf{Mix}(f^*)$ is a subcomplex of $\mathbf{Cond}(S)$ and

$$\mathbf{Pure}(f, y) := \mathbf{Pure}(f^*, y), \qquad \mathbf{Pure}(f) := \mathbf{Pure}(f^*)$$

are defined by 7.6 and are therefore subcomplexes of $\Delta(\mathbf{Cond}(S))$.

The notions of **Pure** and **Mix** are motivated by the following consideration: $c \in \mathbf{Pure}(f)$ iff $\#(f(\mathbf{Sat}(c))) = 1$, i.e., all x satisfying c are "of the same color". Otherwise, $c \in \mathbf{Mix}(f)$ iff there are objects x and y satisfying c with different colors, $f(x) \neq f(y)$.

7.7. Consider the following example of a boolean function f:

$$f = x_1 x_2 x_3 \vee \bar{x}_1 \bar{x}_2 \bar{x}_3 .$$

Let $c_1 c_2$ be the two conditions introduced in 7.2.4. Then $c_1 \in \mathbf{Mix}(f)$ and $c_2 \in \mathbf{Pure}(f)$. To be more precise: $c_2 \in \mathbf{Pure}(f, 0)$. To prove the first we remark that $f(1, 0, 0) = 0$ but $f(0, 0, 0) = 1$. To prove the second we remark that

$$\mathbf{Pure}(f, 0) = \{c : f(\mathbf{Sat}(c)) = 0\}.$$

But $x \in \mathbf{Sat}(c_2)$ implies $\bar{x}_1 \vee x_2 = 0$ and therefore $f(x) = 0$.

8. Topological considerations

8.1. Let K be a simplicial complex. The *geometric realization* $|K|$ of K is by definition the set of all functions p defined over the set of vertices of K with values in the interval $[0, 1] \subseteq \mathbb{R}$ satisfying the following conditions:

(1) $\operatorname{supp} p := \{v \in K : p(v) \neq 0\} \in K$;

(2) $\sum_{v \in K} p(v) = 1$.

Proposition 8.2. *Assume S to be fully faithful. Then for the ith homology group of* $\mathbf{Cond}(S)$ *with coefficients in $\mathbb{Z}$ holds*:

$$H_i(\mathbf{Cond}(S); \mathbb{Z}) = \begin{cases} \mathbb{Z}, & \text{if } i = 0, \\ \mathbb{Z}^t, & \text{if } i = N - 1, \\ 0, & \text{otherwise,} \end{cases}$$

where $t = (m-1)^N$ with $m = \#(V)$, $N = \#(A)$. If $m = 2$ (the case of boolean information systems), then $|\mathbf{Cond}(B_N)|$ is homeomorphic to the $(N-1)$-dimensional sphere. (Graw proved moreover, that in general $\mathbf{Cond}(S)$ is shellable and $|\mathbf{Cond}(S)|$ is a bouquet of t $(N-1)$-dimensional spheres.)

Proposition 8.3. *Let K be a classification problem with underlying information system S. Then $\mathbf{Mix}(K)$ is a subcomplex of $\mathbf{Cond}(S)$, homogeneously $(N-2)$-dimensional. $|\mathbf{Pure}(K)|$ is homotopical equivalent to the complement of $|\mathbf{Mix}(K)|$ in $|\mathbf{Cond}(S)|$.*

Proposition 8.4 (Lefschetz Duality). *If $S = B_{N+1}$ then $\mathbf{Pure}(K)$ and $\mathbf{Mix}(K)$ are connected by the following isomorphism*:

$$H_i(\mathbf{Pure}(K); F) = H_{N-i}(\mathbf{Cond}(S), \mathbf{Mix}(K); F).$$

F is any field of characteristic zero.

This theorem allows to compute the homology groups $H_i(\mathbf{Pure}(K), F)$ knowing $H_i(\mathbf{Mix}(K); F)$ and vice versa. To do this use the exact homology sequence

$$\cdots \to H_{i+1}(\mathbf{Cond}(S), \mathbf{Mix}(K); F) \to H_i(\mathbf{Mix}(K); F)$$
$$\to H_i(\mathbf{Cond}(S); F) \to H_i(\mathbf{Cond}(S), \mathbf{Mix}(K); F) \to \cdots$$

and take into account Proposition 8.2.

Let

$$h_i(\mathbf{Pure}(K)) := \dim_F H_i(\mathbf{Pure}(K); F),$$
$$h_i(\mathbf{Mix}(K)) := \dim_F H_i(\mathbf{Mix}(K); F)$$

be the Betti numbers of $\mathbf{Pure}(K)$, $\mathbf{Mix}(K)$, respectively. Then Propositon 8.4 yields the following:

Corollary 8.5. *Assume $N \geqslant 3$. Under the assumptions of Proposition 8.4 the following equalities hold:*

$$h_0(\mathbf{Pure}(K)) = h_{N-2}(\mathbf{Mix}(K)) + 1,$$
$$h_i(\mathbf{Pure}(K)) = h_{N-2-i}(\mathbf{Mix}(K)) \quad \text{if } N-2 > i > 0,$$
$$h_{N-2}(\mathbf{Pure}(K)) = h_0(\mathbf{Mix}(K)) - 1,$$
$$h_i(\mathbf{Pure}(K)) = 0 \quad \text{if } i > N-2.$$

The following result gives evidence that classification problems which are difficult from the topological point of view are computational intractable.

Theorem 8.6. $\mathrm{Size}(K) \geqslant (h_0(\mathbf{Pure}(K)) - 1)/(m-1)$.

Corollary 8.7. *If $m = 2$, then*

$$\mathrm{Size}(K) \geqslant h_0(\mathbf{Pure}(K)) - 1 = h_{N-2}(\mathbf{Mix}(K)),$$

i.e., classification problems with many "$(N-2)$-dimensional holes" are of high complexity.

9. The Euler–Poincaré characteristic

9.1. Let K be a finite simplicial complex. The alternating sum

$$\chi(K) := \sum_{i=1}^{\infty} (-1)^i \#\{s \in K : \dim s = i\} = \sum_{s \in K} (-1)^{\dim s},$$

i.e. the number of simplexes of even dimension minus the number of simplexes of odd dimension is a topological invariant because

$$\chi(K) = \sum_{i=0}^{\infty} (-1)^i \dim_F H_i(K; F)$$

where F is an arbitrary field. This invariant is called the Euler–Poincaré characteristic. For a poset P one defines $\chi(P) := \chi(\Delta(P))$. The Lefschetz Duality 8.4 has the following straightforward consequence:

Proposition 9.2. *Let C be a classification problem over B_N, the boolean information system of order N. Then*

$$\chi(\mathbf{Mix}(C)) - 1 = (-1)^N (\chi(\mathbf{Pure}(C)) - 1).$$

Proposition 9.3. *The Euler–Poincaré characteristic of* GAP(n) *satisfies the following properties*:

(1) $\chi(\mathbf{Pure}(\mathbf{GAP}(n),0))$

$$= h_0(\mathbf{Pure}(\mathbf{GAP}(n),0))$$

$$= h_0(\mathbf{Pure}(\mathbf{GAP}(n))) - 1 = \Omega(n^n(n-2)!),$$

(2) Mix(GAP(n)) *is sheliable* (*this result is due to Graw*) *and therefore*:

$$\chi(\mathbf{Mix}(\mathbf{GAP}(n))) = 1 + h_2(n-2)(\mathbf{Mix}(\mathbf{GAP}(n))).$$

For $n=3$ *one gets* $\chi(\mathbf{Pure}(\mathbf{GAP}(3))) = 13$.

Combining this proposition with Corollary 8.7 one gets again Corollary 5.6.

10. Lower bounds

Proposition 10.1. *Let* $S=(X, A, V, \rho)$ *be an information system which is assumed as always to be fully faithful. Let* $\Gamma = (\Gamma, E_a, E_b, o, t)$ *be an arbitrary arsenal and let* $\Phi : \Gamma(S) \to \Gamma$ *be a homomorphism of the arsenal belonging to* S (*see* 6.8) *into* Γ. *Then*

$$I(\Phi(\operatorname{Ker} f)) \leq \operatorname{size}(f).$$

This follows directly from 6.9 and the fact that $I(\Phi(x)\,|\,\Phi(y)) \leqslant I(x\,|\,y)$. This proposition yields a universal strategy for proving lower bounds for the size of classification problems: To get lower bounds one has to construct an adequate arsenal Γ and a homomorphism of $\Gamma(S)$ into Γ. Assume this is possible and assume further that $I(\Phi(\ker f))$ can be computed, then this number is a lower bound for size(f). We will give an example of this strategy in this section.

10.2. Let A be the quotient ring of the (commutative) polynomial ring $\mathbb{Z}[t_0, t_1, t_2, \ldots]$ over the ring $\mathbb{Z}$ of integers in enumerable many variables by the ideal generated by all quadratic polynomials $t_i(t_i+1)$. Let R be the following binary relation over A:

$f(t_0, t_1, \ldots) R g(t_0, t_1, \ldots)$ if either one of the following conditions is satisfied:

(1) $g(t_0, t_1, \ldots) = f(t_0, t_1, \ldots, t_{i-1}, t_i, t_{i+1}, \ldots, t_{j-1}, t_i, t_{j+1}, \ldots)$,

(2) $g(t_0, t_1, \ldots) = f(t_0, t_1, \ldots, t_{i-1}, t_j, t_{i+1}, \ldots, t_{j-1}, t_i, t_{j+1}, \ldots)$,

i.e., g can be obtained from f by identifying variables or by transposing variables. Let R^* be the transitive closure of R. Let $[f]$ be the equivalence class

$$[f] := \{g : fR^*g \text{ and } gR^*f\}.$$

$[f]$ consists of all polynomials which arise from f by permutations of variables. R^* defines a partial order $\leqslant$ in the set $\bar{A}$ of all equivalence classes $[f]$:

$$[f] \leqslant [g] \text{ iff } fR^*g.$$

To every natural number n we consider the following finite subposet $\bar{A}_n$ of $\bar{A}$: Let $f_n(t_0, t_1, \ldots, t_{2^n-1})$ be the following polynomial:

$$f_n(t_0, t_1, \ldots, t_{2^n-1}) := \sum t_I$$

where the sum is running over all $t_I = \prod_{i\in I} t_i$, for which there is a $j \in \{1, 2, \ldots, n\}$ such that the jth bit in the binary representation of all $i \in I$ coincides. Now define $\bar{A}_n := \bar{A}_{\geq [f_n]}$ (see 6.1).

10.3. $\bar{A}_n$ is a finite poset. Following 6.3.1, $\bar{A}_n$ defines an arsenal a_n the underlying poset of which is $\bar{A}_n$. E_a, the set of amalgamations, can be identified with the set of all $([f], [g])$ where g can be obtained from f by identifying two variables, say t_i and t_j. E_b, the set of branchings, consists of pairs $([g], [f])$, such that the inverse amalgamation $([f], [g])$ satisfies the following property: If g is obtained from f by identifying t_i and t_j, then the coefficients of t_i and t_j in f are both equal to 1.

Theorem 10.4. *a_n is an arsenal and there is a homomorphism of arsenals*

$$\Phi : b_n \to a_n.$$

This homomorphism can be constructed as follows: Let c be an element of b_n and let $C = (B_n, f)$ be a classification problem with $\ker f = c$ (see 6.8). Without restriction of generality we can assume that $\operatorname{Im} f = \{1, 2, \ldots, m\}$. Let $\overline{\mathbf{Pure}}(f, i)$ be the closure of $\mathbf{Pure}(f, i)$ in $\mathbf{Cond}(B_n)$. Consider the following polynomial:

$$\varphi(f) := \sum_{i_1 < i_2 < \cdots < i_k} \chi(\overline{\mathbf{Pure}}(f, i_1) \cap \overline{\mathbf{Pure}}(f, i_2) \cap \cdots \cap \overline{\mathbf{Pure}}(f, i_k)) t_{i_1} t_{i_2} \cdots t_{i_k}.$$

Then Φ defined by $\Phi(\ker f) := [\varphi(f)]$ is the desired homomorphism of arsenals.

Corollary 10.5. $I(\Phi(\ker f)) \leq \operatorname{size}(f)$.

Acknowledgment

This work was written when the author was in 1986 on leave at the Laboratoire Informatique Théorique et Programmation, Université Paris VI. I would like to thank J. Berstel, J. Sakarovitch, Ch. Reutenauer and many others for their hospitality. The definition of the boolean arsenal b_n, being the starting point for a general theory of arsenals, was the result of a discussion with B. Graw at the Banach-Center 1985 in Warsaw.

References

[1] L. Budach, Two pebbles don't suffice, in: Lecture Notes in Computer Science **118** (Springer, Berlin, 1981) 578–589.

[2] L. Budach, Information und Rechnen, in: *Zur Bedeutung der Information für Individuum und Gesellschaft; Berichtsband der wissenschaftlichen Konferenz zum Leibniz-Tag der Akademie der Wissenschaften der DDR* (1983) 191–208.

[3] L. Budach, Klassifizierungsprobleme und das Verhältnis von deterministischer zu nichtdeterministischer Raumkomplexität, *Seminarberichte* **68** (1985) 1-64.
[4] L. Budach, A lower bound for the number of nodes in a decision tree, *Elektron. Inform. verarb. Kybern.* **21** (1985) 221-228.
[5] L. Budach and B. Graw, Nonuniform complexity classes, decision graphs and homological properties of posets, in: Lecture Notes in Computer Science **208** (Springer, Berlin, 1985) 7-13.
[6] L. Budach, The complexity of classification problems, Banach Center Publications, 1985.
[7] R. Godement, *Topologie Algébrique et Théorie des Faisceaux* (Hermann, Paris, 1958).
[8] J. Kahn, M. Saks and D. Sturtevant, A topological approach to evasiveness, *Combinatorica* 4 (1984) 297-306.
[9] W. Marek and Z. Pawlak, Information storage and retrieval systems: Mathematical foundations, *Theoret. Comput. Sci.* **1** (1976) 331-354.
[10] C.F. Picard, *Theorie der Fragebögen* (Akademie-Verlag, Berlin, 1973).
[11] P. Pudlák and S. Zâk, Space complexity of computations, Manuscript (1983) 1-30.
[12] W. Savitch, Relations between nondeterministic and deterministic tape complexities, *J. Comput. Syst. Sci.* 4 (1970) 177-192.
[13] E.H. Spanier, *Algebraic Topology* (McGraw-Hill, New York, 1966).
[14] R.P. Stanley, *Combinatorics and Commutative Algebra* (Birkhäuser, Boston, MA, 1983).
[15] H. Thiele, Eine graphentheoretische Realisierung von Retrieval-systemen und äquivalente Umformungen derartiger Realisierungen, Manuscript, Humboldt-Universität, Berlin, 1976.